United States Patent Application

of

Terrance C. Slattery

Shawn G. Coville

for

Methods and Apparatus for

Computer Based Training Relating to Devices

RELATED APPLICATIONS

The following applications contain subject matter related to the subject matter of the present application, are assigned to the assignee hereof and have been filed on the same date as the present application.

1. Terrance C. Slattery, Shawn G. Coville, Gregory R. Long, M. Tylor Burton III, "Methods and Apparatus for a Virtual Classroom."

2. Terrance C. Slattery, Shawn G. Coville, Gregory R. Long, M. Tylor Burton, Lawrence F. Galvin, "Methods and Apparatus for Computer Based Assessment Relating to Devices."

3. Terrance C. Slattery, Shawn G. Coville, Gregory R. Long, Karl J. Schmidt, Kenneth C. M. Stevenson III, Mark W. Decker, "Methods and Apparatus for a Learning Structure in Computer Based Training Relating to Devices."

4. Terrance C. Slattery, Shawn G. Coville, Karl J. Schmidt, Kenneth C. M. Stevenson III, "Methods and Apparatus for a User Interface for Computer Based Training Relating to Devices."

BACKGROUND OF THE INVENTION

The present invention relates to computer based training, and more particularly, to methods and apparatus for computer based training providing hands-on training related to devices.

Traditionally, students have been taught about devices in a classroom setting where the students were lectured about the devices and presented with written tests. Although students have been taught in a lab setting, this requires students to travel to a specific location to perform the lab Thus, making it difficult and inconvenient for students without easy access to a lab.

Thus, it has been difficult for students to gain hands-on experience regarding devices in a classroom setting, in their office place, or at home. Further, the need for more hands on time with devices is becoming critical for professionals that are working towards certifications such as the CISCO Certified Internetworking Expert (CCIE) certification, which is the highest certification level bestowed by CISCO.

Accordingly, there is a need for a distance based solution that provides students with hands-on training with devices.

SUMMARY OF THE INVENTION

Accordingly, the present invention is directed to methods and apparatus that substantially obviates the above and other disadvantages of the prior art.

Methods and systems consistent with the present invention include a system for training a user regarding controlling a device. This system includes a user computer for accepting device control information regarding controlling the device for the purposes of training, and a device controller remotely connected to the user computer. The device controller preferably includes

2

3

means for receiving the device control information from the user, and means for transferring the device control information to the device.

In another aspect, the invention comprises a method for training a user regarding controlling a device. This method includes a user remotely connecting to a device controller using a user computer; the user computer transferring user information regarding control of the device to a device controller, and the device controller transferring the user information to the device.

The summary of the invention and the following detailed description should not restrict the scope of the claimed invention. Both provide examples and explanations to enable others to practice the invention. The accompanying drawings, which form part of the description for carrying out the best mode of the invention, show several embodiments of the invention, and together with the description, explain the principles of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

In the Figures:

Figure 1 illustrates a block diagram of a computer-based system for training of devices in accordance with methods and systems consistent with the invention;

Figure 2 illustrates a block diagram showing a distributed set of pod controllers, in accordance with methods and systems consistent with the invention;

3

4

Figure 3 provides a detailed diagram of a pod controller, in accordance with methods and systems consistent with the invention;

Figure 4 illustrates a graphical interface, in accordance with methods and systems consistent with the invention;

Figure 5 provides a flow diagram for a computer based training system, in accordance with methods and systems consistent with the invention;

Figure 6 illustrates an example page that may be displayed to a user to log in or to request account information, in accordance with methods and systems consistent with the invention;

Figure 7 illustrates a page that may be used for setting up a user account, in accordance with methods and systems consistent with the invention;.

Figure 8 illustrates Relay Control Boxes, in accordance with methods and systems consistent with the invention.

Figure 9 illustrates a block diagram of a pod controller including a wiretap, in accordance with methods and systems consistent with the invention;

Figure 10 provides an illustration of a Virtual Classroom, in accordance with methods and systems consistent with the invention;

Figure 11 provides a flow diagram for the virtual classroom, in accordance with methods and systems consistent with the invention;

Figure 12 illustrates a logical diagram of a skills learning module, in accordance with methods and systems consistent with the invention;

5

Figure 13 provides a logical description of information that may be stored by the assignment and story sub-module, in accordance with methods and systems consistent with the invention;

Figure 14 illustrates an example of an assignment and story sub-module, in accordance with methods and systems consistent with the invention;

Figure 15 illustrates an example of a page that provides a user with a suggested approach, in accordance with methods and systems consistent with the invention;

Figure 16 illustrates a template for a page that provides a user with a sample solution, in accordance with methods and systems consistent with the invention;

` Figure 17 illustrates an example of an html page that provides a sample solution, in accordance with methods and systems consistent with the invention;

Figure 18 illustrates an example of a page for the check results module, in accordance with methods and systems consistent with the invention;

Figure 19 illustrates an example of a page for the check results module, in accordance with methods and systems consistent with the invention.

Figure 20 illustrates an example of a page that includes archived information for a lab, in accordance with methods and systems consistent with the invention.

Figure 21 provides an illustration of an assessment engine that uses a wiretap to check the configuration of the devices, in accordance with methods and systems consistent with the invention; and

Figure 22 illustrates an embodiment in which the server and controller contains an assessment engine, in accordance with methods and systems consistent with the invention.

DETAILED DESCRIPTION OF THE INVENTION

Reference will now be made in detail to embodiments of the invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

Engine

~~Figure 1 illustrates a simple block diagram of a computer-based system for training~~ relating to devices. As shown, the system preferably includes customer premise equipment 12, a communications link 14, a firewall 16, a communications link 18, a server and controller 20, a database 22, a pod controller 24, and a pod 26. The Customer Premise Equipment (CPE) 12 preferably includes a computer 28 provided with a browser program 30 and a network application program 32. The browser 30 is preferably a browser for Internet/Intranet communications, such as a Netscape Navigator™ browser or a Microsoft Internet Explorer™ browser. The network application program 32 may be a program such as TELNET. The communications link 14 preferably traverses the Internet or an Intra-net. The pod controller 24 may control one or more pods 26 each of which may contain one or more user devices 40_1 to 40_3. In one embodiment, the user devices 40_1 to 40_3 are network equipment, such as CISCO

6

7

~~type switches or routers. In other embodiments, these user devices may be Programmable Logic~~ Controllers (PLCs), Chemistry Equipment, or any other type of device. Further, a pod controller 24 may also control one or more infrastructure devices (not shown). These infrastructure devices ~~provide an authentic environment for which a real-world scenario may be written.~~

C.H. 7/27/00
INS CSA

Figure 2 illustrates a simple block diagram of an embodiment with multiple pod controllers 24_1 to 24_3. In this embodiment, pod controllers 24_1 and 24_2 are behind firewall 16_1, and pod controller 24_3 is behind firewall 16_2.

Figure 3 provides a more detailed diagram of the pod controller 24, consistent with an embodiment of the invention. The pod controller 24 may include a device control module 302, a user communications module 304, a mentor communications module 306, an infrastructure control module 308, a device communications, control, and multiplexor module 310, and an interface to device module 312. The pod controller 24 may be connected to user devices 314, and infrastructure devices 316.

INS C

CH
C4

~~The device control module 302 is used to control user accessible devices. It incorporates~~ the control software that enables the pod control system to load starting configurations into the user devices, reset the user devices, and save final configurations. The control within this module preferably is high level and generic across all devices, increasing the modularity and maintainability of the overall system.

The user communications module 304 operates such that when a user connects to a user

C.H. 7/27/00

through
device, the connection is made ~~though~~ the user communications module. This module receives the connection from the network and validates the user's authorization to access specific devices.

7

8

This module further translates information received by a user in one protocol into a protocol for feeding into the user device.

The mentor communications module 306 permits a mentor to monitor and participate in controlling the user devices during a learning exercise. The mentor communications module 306 authenticates and authorizes the mentor to connect to specific devices through a computer. A mentor may be a program, such as an Artificial Intelligence program, a person, or any type of hardware or software capable of aiding a user in learning about the user device and its operation.

~~The infrastructure control module 308 allows additional devices to be interconnected to~~ the user devices in order to replicate real-world scenarios. These devices are part of the infrastructure and require separate control by the pod controller. As such, this module provides the control of the infrastructure devices that are needed to create a real-world scenario for the user. The infrastructure devices 316 are discussed in greater detail below. Further, this module, in conjunction with device, communications, control and multiplexer modules 310, permits devices to be moved between pods connected to the pod controller 24. This will be discussed in further detail below.

A device communications, control, and multiplexor module 310 provides low-level communications and control for each device. In addition, this module provides a mechanism for multiple modules to simultaneously communicate with a single device. Further, as discussed above, this module permits devices to be moved between pods connected to the pod controller 24. For example, Figure 2 illustrates user device 40_4 as part of pod 26_2. If a user wishes to perform an assignment that needs four user devices, the device communications, control, and

8

9

multiplexor module 310 for user device 40_4 may be reconfigured to move user device 40_4 into pod 26_1. Further, the infrastructure control module 308 may also be reconfigured so that the infrastructure devices necessary for supporting this move are also reconfigured. Thus, pod 26_1 would now include four user devices so that a user wishing to perform the assignment can use pod 26_1 and pod controller 24_1.

The interface to device module 312 is generally implemented in a combination of hardware and software, and provides the pod controller 24 with the capability of the pod controller 24 to communicate directly with the device. For example, if the device 314 uses an RS-232 or similar interface, this module may include software and hardware for implementing an RS-232 connection to the device. Other connection types are possible, such as Ethernet, and the connection type will be dependent on the specifics of the device. Further, this module is capable of receiving information from a user in an application layer protocol, such as TELNET, and translating it so that the information can be communicated directly to a user device.

Each of the modules may be implemented by software running on a processor or computer. As will be obvious to one skilled in the art, all of these modules may be implemented on a single processor, or multiple processors may be used. A variety of computer programming languages may be used to implement the software programs.

The infrastructure devices 316 permits user devices 314 to be interconnected into realistic configurations. The infrastructure devices are not directly accessible by the user. However, the system may implement functions that affect the configuration of these devices depending upon the user's actions. For example, a Frame Relay switch may be used to

10

implement a wide area network connection between user devices. Such a switch may need to be configured and controlled for specific exercises, but because multiple users may need to share it, the switch will not be directly available to the user.

The server and controller 20 may be a computer including memory, a processor, a communications port, a display screen, keyboard, and a mouse. Further, the server and controller 20, for example, may be a web server, such as an Apache or an AOL Server, that uses the Common Gateway Interface (CGI) to allow programs to be executed within the server and controller 20. These programs may provide services to a user via the web such as static web pages, a scheduling mechanism to allow a user to schedule the execution of a specific lab or course at a given time, a lab or course catalog displayed via a web page showing all labs or courses which can be executed, an electronic commerce engine through which users may purchase labs or courses online, and/or a web page which provides a graphical interface for accessing the user devices in specific labs and initiating device control software.

The server and controller 20 may directly provide static web pages or they may be dynamically created from templates and other data contained within a system database. A scheduling component may interface to a database of reservations to determine the availability of devices for a lab. Further, the scheduling component may have the capability to add, delete, or change a user's schedule. A course catalog may include a nearly static page generated from a database of existing labs or courses. This page may be displayed to a user as an html web page. Further, the server and controller 20 may connect to one or more systems, such as electronic commerce systems, training/student management systems, personnel records systems, etc. This

10

~~connection may be over an Internet, or such systems may reside within the server and controller~~ 20. An electronic commerce system is a system for conducting ~~commerce such as on-line ordering and accepting credit payments over an Internet.~~

The server and controller 20 may send either static or dynamic web pages to the user's CPE 12 so that they may be displayed to the user through their browser. The content of the user interface page may contain buttons and hot links for the user to invoke the device-specific operations necessary to control the user devices for the purposes of accomplishing a lab or course.

The user computer 28 may be a standard user computer such as an IBM compatible Personal Computer (PC) or Macintosh. Further, the user computer may include a memory, a processor, a keyboard, a mouse, and a port for receiving and transmitting information. The keyboard and mouse may be used by a user to enter information into the computer. The memory may be used for storing programs, such as the browser and network applications programs. The processor may be used for executing these programs. The display screen may be used for displaying information to the user such as html pages, and screens for the browser and network applications programs. The communications port may be used for sending and receiving information over a communications link.

Figure 5 provides a flow diagram for the computer based training system of Figure 1. A user initializes the system by instructing the browser 30 on his/her computer 28 to connect to the server and controller 20 (S502). The firewall 16 is preferably set up to allow the browser 30 and server and controller 20 to freely communicate. The server and controller 20 preferably displays

~~a page to the user requesting an account identification and password and then uses this~~ information to determine if the user has an account (S504). Figure 6 illustrates an example html page 600 that may be displayed to a user to request a user's account identification 602 and password 604. The user can then click on the submit button 606 to submit the information once it has been entered. The user account may contain, among other things, a course for the user and course specific information. If the user has an account, the course is selected and started (S506). If the user does not have an account, he/she is directed to a page for setting up a user account (S518). For example, as illustrated in Figure 6, a user without an account identification is directed to click on a Register button 608. Figure 7 illustrates an html page 700 that may be used for setting up a user account. This page may request, for example, his/her name 702, address ~~704, a user name 706, and a password 708.~~

After the course is selected, the server and controller 20 sends an instruction to the pod controller to initialize a pod 26 associated with the selected course (S508). Further, the server and controller 20 sends an instruction to the firewall 16 to permit the user's CPE 12 to access the pod controller 24 (S510). The user then connects to the pod controller 24 using a network application program 32, such as Telnet (S512). Through this mechanism, the user can exercise control over the user devices and perform the assigned training exercise.

A graphical interface may be presented to the user that includes buttons and/or hot links regarding the various user devices. These buttons and/or hotlinks may be used by the user to perform various functions regarding the user devices. The graphical interface may include a button for resetting a user device to its initial starting configuration, a button for saving an

13

existing configuration, a button for loading a new configuration, or a button for forcing the user device into a diagnostic control mode. The user may perform these functions by clicking on a specifically labeled button on the graphical interface. The specific buttons or hotlinks and their respective functions will depend on the specifics of the user devices that make up the lab or course. For example, a chemistry lab may use buttons for controlling a mechanical shuttle which moves a beaker to different locations within an experiment, the addition of a chemical to the beaker, controlling a heating element, or controlling the agitation of the solution within the beaker.

Figure 4 provides an example of a graphical interface a user may see when interacting with a lab including three Cisco routers. The available functions in this example are: reset 402, save configuration 404, load configuration 406, and cycle-break 408. The user may click on the icon 410 of a user device to open a communications connection to the user device when completing the objectives of the lab or course. Further, in this example, the user may click on links for the instructions 412, scenario and assignment 414, mentoring 416, view plan 418, suggested approach 420, sample solution 422, and check results 424. By clicking on one of these links, the user will be taken to an associated web page. These links will be discussed in greater detail later.

After completion of the course, the user's CPE 12 is disconnected from the user devices 40_1 to 40_3 by the pod controller 24, and the user devices are reset to their original settings (S516). For routers, these user devices can be reset by sending a specific sequence to the router that permits the system to gain access to the router even if the user alters the passwords for the

/14

router. This may be accomplished by turning off the router, and then turning it back on. During the boot up, control characters can be sent to the router to cause it to enter ROM mode. Once the router is in ROM mode, the user device can be reset so that another user can use the user devices regardless of what the previous user did to the user device.

If the user device is a CISCO-type switch, a specific button needs to be pushed during power up in order to enter ROM mode. This switch may be replaced by a relay to close the switch. Thus, the pod controller could reset this user device by closing the relay, or by manually operating the switch. As an example, Figure 8 illustrates three Relay Control Boxes 802, 804, and 806 for resetting devices in which a physical switch needs to be pressed at a particular point in time. Each Relay Control Box may include eight internal relays for controlling eight independent devices. Each relay replaces the physical switch in the device. As illustrated, a master Relay Control Box 802 is connected directly to the Pod Controller 24, and each client Relay Control Box 804 and 806 is serially connected to the master Relay Control Box 802. When a device is to be reset, the pod controller 24 sends an address for the device to the Relay Control Boxes 802, 804, and 806. The Relay Control Boxes 802, 804, and 806 then use this address to close a relay associated with this address that in effect closes the switch for the device. Thus, the Relay Control Boxes and Pod Controller can individually open and close the switches for the particular devices. This example is provided as merely one possible implementation, and as will be obvious to one of skill in the art numerous other implementations are possible. Further, the specifics of how to reset a device will depend on the particulars of the specific device.

15

to the pod controller through a COM port. In this embodiment, the pod controller 24 converts the user information from the application layer format it is received in (for example, TELNET) to a format that can be sent to the router through the router's COM port. This is preferably accomplished by the user communications module 304 of the pod controller 24.

Figure 9 illustrates a block diagram of a pod controller 24 including a wiretap 902 , according to an embodiment. As shown, the pod controller of this embodiment is connected to a user's network application program 32, such as TELNET, over a network connection 904 that preferably includes, referring back to Figure 1, a communications link 14 from the CPE 12 to the firewall 16 and a connection from the firewall 16 to the pod controller 24 over communications link 18. Also connected to the wiretap 902 is a mentor's equipment 906 operating a network application 908, such as Telnet. Through this mechanism, a mentor can monitor the instructions the user is sending to the user devices in the pod. As such, the mentor can "watch" what the user is doing, so that the mentor can ensure that the user is properly controlling the user device. Various software and hardware can be used to implement the wiretap. For example, KIBITZ, a program available with the freeware package EXPECT, allows two users to collaborate over a network while interacting with a single program. Thus, by using one KIBITZ for each user device, everything the user types can be seen by the mentor, and visa versa. Other programs with similar functionality may also be used to achieve this type of operation. The wiretap 902 may be implemented in the device communications, control, and multiplexor module 310 of the pod controller 24.

Additionally, this wiretap may permit the mentor to take control of the user devices in the pod so that the user can "watch" the instructions the monitor is sending to the user devices. Thus, the mentor can show the user how to correctly operate the user devices to perform a task. In another embodiment, the wiretap may be connected to an Artificial Intelligence program that oversees the user's operation of the user devices.

Virtual Classroom

Figure 10 provides an illustration of a Virtual Classroom, 1000 consistent with an embodiment of the invention. As shown, the Virtual Classroom 1000 preferably includes a plurality of student terminals 1010_1 to 1010_16, an instructor terminal 1012, a communications link 1014, a firewall 1018, a server and controller 1020, a pod controller 1024, and a pod 1026. The pod 1026 preferably includes a plurality of user devices 1040_1 to 1040_3. The number of devices needed will depend on the class objectives. The server and controller is optionally connected to a database 1022 that stores class specific information. The student terminals 1010_1 to 1010_16 each preferably include a browser (not shown) that may be a browser for Internet/Intranet communications, such as a Netscape Navigator™ browser or a Microsoft Internet Explorer™ browser. Further, each student terminal 1010_1 to 1010_16 preferably includes a network application (not shown) for sending control information to the user devices. In one embodiment, the network application program is TELNET. The communications link can be any type of communications link, and may be, for example, a link across an Internet or Intra-net. Further, a server (not shown) may be used to connect the student and instructor terminals to

an Internet or Intra-net that links the terminals to the server and controller 1020 and pod controller 1024.

Individual students need not be located at the same physical location. For example, one student may be located at one physical location, another student may be located in a physical location in a different city, a third student may be located in yet another city, and so on. The communications between the instructor and students can be supported using a variety of communications applications. For example, ILINC, Placeware, or Caliver communications applications may be used for supporting these communications. Further, the class can be conducted synchronously or asynchronously.

Figure 11 provides a flow diagram for the virtual classroom, consistent with the invention. An instructor initializes the class session by sending information from the instructor terminal 1012 to the server and controller 1020 (S1102). The server and controller 1020 then sends initialization information to the pod controller 1024 (S1104). The pod controller then resets the user devices 1040_1 to 1040_3 (S1106). The server and controller 1020 then instructs the firewall 1018 to permit the student terminals to access the pod controller 1024 (S1108). Through this system and mechanism students can be trained regarding controlling a user device by exercising actual control over the user devices in the pod (S1110).

After the class ends, the instructor can instruct the server and controller 1020 to save the configurations of the user devices 1040_1 to 1040_3 to allow the class to pick up where it left off at a later time. The user devices are then preferably reset to permit another class to use the user devices (S1112).

17

18

Organization of Learning Structure

Traditionally, students are taught using a learning structure that involves an instructor first lecturing or telling students about a subject, such as how to operate a device. Next, the instructor shows the students how to perform the task, and finally students are provided with a task and asked to perform it.

In an embodiment of the invention, rather than first instructing the students regarding the operation of a device, students are invited to try and perform a task regarding the device. If the student is unable to perform the task they are provided with a first level of help. This first level of help includes clues or hints regarding performing the task and top-level information. If the student needs further help, he/she is provided with a second level of information. This second level of information includes detailed information regarding performance of the task, and in one embodiment includes showing the user the exact steps to take in order to perform the task.

Figure 12 illustrates a logical diagram of a skills learning module 1200 consistent with the invention. The skills learning module 1200 may include module metadata 1210, an assignment and story module 1220, a suggested approach module 1230, a sample solution module 1240, and a check results module 1250. In an embodiment, the skills learning module is stored in database 22 along with each of its sub-modules. Each of these sub-modules may be presented to a user in the form of a web page in an html format. By structuring the data in a hierarchical manner, the user may view only that information which is needed at a particular time.

Module metadata 1210 is a set of data that describes the learning module, and may

include information regarding the learning environment for the module, the author of the module,

the difficulty level, target audience, and other like information. For example, module metadata

1210 may include the assignment's name, its difficulty, the expected time to complete it, its

authorship, and the skills or knowledge that it exercises.

The assignment and story sub-module 1220 stores information regarding the context of an

assignment so that a user may frame the learning experience in the real world. Figure 13

provides a more detailed logical description of information that may be stored by the assignment

and story sub-module 1220. As illustrated the assignment and story sub-module 1220 may

include a details sub-module 1310, an assignment sub-module 1320, a story sub-module 1330, a

conditions sub-module 1340, a notes sub-module 1350, and a diagram sub-module 1360. Figure

14 illustrates an example of an assignment and story sub-module that may be presented to a user

in the form of an html web page.

The details sub-module 1310 provides module metadata that is useful to the user. For

example, as illustrated in Figure 14, the details sub-module may include data regarding the title

of the assignment, the technology involved, the level of difficulty, the amount of time to

complete the task, the certification it may be used for, the desired learner outcome, and the

desired network outcomes.

The assignment sub-module 1320 provides a brief textual description of the environment

in which the assignment takes place. For example, as illustrated in Figure 14, the assignment

19

20

sub-module 1420 describes the assignment as designing an Appletalk numbering plan and enabling Appletalk routing.

The story sub-module 1330 provides a detailed textual description of the environment in which the assignment takes place. It typically mirrors a real-world scenario. For example, as illustrated in Figure 14, the story sub-module 1430 describes the story for the assignment as: "Your network manager has told you that your network will soon have to carry Appletalk traffic. In order for this to happen you must plan an Appletalk numbering scheme and assign Appletalk zone names for each of the segments in your network. You will also enable Appletalk routing on all of the active interfaces on your routers. Once Appletalk is enabled on the routers and configured on the interfaces, you should verify that Appletalk is functioning properly."

The conditions sub-module 1340 provides supporting information. For example, as illustrated in Figure 14, the conditions sub-module 1440 may include information such as:

"IP routing is already up and running on this network. DO NOT CHANGE ANY OF THE IP ROUTING CONFIGURATIONS.

Your Apple administrator has given you the following range of Appletalk network numbers, 2000-2999. You may use any number with that range to assign a unique Appletalk network number to each segment in the network. All of the serial links should be configured in the 'cereal zone.' You should make up unique zone names for each of the Ethernet interfaces."

20

$2/$

The notes sub-module 1350 provides exceptions to real-world implementations or things that should be taken into consideration in the performance of the module. For example, as illustrated in Figure 14, the notes sub-module states that the exceptions to real world conditions for this assignment are that the serial links between routers are implemented via direct connections in this lab and do not actually connect through any leased line services for the serial links.

The diagram sub-module 1360 provides a picture or visual of the module's starting equipment. For example, as illustrated in Figure 14, the diagram sub-module 1460 illustrates three interconnected routers.

The suggested approach sub-module 1230 (Fig. 12) provides a high level description of an approach that the learner could take to accomplish the module's assignment. Figure 15 illustrates an example of a page that provides a user with a suggested approach. As illustrated, in this example, the suggested approach is for the user to: "Figure out the Appletalk numbering plan; assign a unique Appletalk cable range to each network segment; note the Appletalk zone names on each network; enable Appletalk routing on the routers; configure the appropriate Appletalk zone names on each active router interface; then verify proper Appletalk operation using show commands."

The sample solution sub-module 1240 (Fig. 12) contains a subject-matter expert's solution to the module's assignment. Figure 16 illustrates a template for a page that provides a user with a sample solution. Each solution can be broken down into a series of tasks 1602. The initial display may only show each task, along with a one-line description for the task. The user

21

may then click on a task to expose more detailed information regarding the tasks, such as hints 1604 and steps 1606. A hint 1604 is designed to assist the user in accomplishing the tasks and is useful for users who need some initial direction on how to get started with the task, but do not need detailed information on the process. A step 1606 is a one line description that may include an action to take, the result of that action, and an explanation on why the step was necessary and the result that was achieved. In another embodiment, the sample solution may show all the details of the sample solution. Figure 17 illustrates an example of an html page that provides such a sample solution.

The check results sub-module 1250, provides user assessment because during the course of performing a lab or course, a user may wish to check his or her final results. The user's results may be assessed in a variety of ways. For example, this assessment may be performed by examining the final configuration of the devices as compared to a final configuration produced by a subject matter expert. Figure 18 illustrates an example of a page that provides a user with one possible set of final device configurations which can be used to compare against the results obtained by the user. In another embodiment, the check results module may provide a set of steps that the user can execute to assess his or her final configuration. For example, in a Cisco router lab, the user may be told to how to examine the routing table of a specific device and look for specific outputs. Figure 19 illustrates an example of a page that provides a user with steps for checking their results.

Further, referring back to the example illustrated in Figure 4, the user may select from the graphical interface illustrated to view instructions regarding the lab 412. By clicking on this link,

23

a page is displayed to the user regarding the general instructions for performing the lab. Also

shown in the example illustrated in Figure 4 is a link to view plan 418. In this example, prior to

beginning the lab, the user is presented with a series of predetermined questions designed to help

the user structure his/her lab approach, thus allowing the user to plan his/her approach before

beginning the lab. These questions may include, for example, questions regarding the user's plan

of attack, the impact the user thinks his/her approach will have on the rest of the network and any

potential trouble areas and anticipated challenges, and how he/she expects the lab to be helpful.

Further, these questions may be predetermined questions or be adaptive questions that depend

upon the user's answers to the questions. When the user later clicks on view plan 418, a web

page is displayed to him/her that includes answers to these questions.

After a user completes a lab, in an embodiment, the user is led through a debrief

question/answer section designed to reinforce the lessons taught by the lab and help the user

identify ways that he/she might incorporate these skills in a production environment. The user's

answers are then saved for future reference. These questions may include, for example,

questions regarding how the user felt his/her plan worked out, the impact his/her approach might

have had in a production environment, how he/she might change their approach, how he/she

thinks his/her revised approach might work in a production environment, what he/she learned,

and how the lab was helpful to him/her. Further, these questions may be predetermined

questions or be adaptive questions that depend upon the user's answers to the questions.

In one embodiment, after completion of a lab, the final configurations of the user devices

along with other information are archived so that the user may retrieve the information and

23

$\mathcal{24}$

configurations at a later time. Figure 20 illustrates an example of a page that includes archived results after a user completes a lab where the user devices are Cisco type routers. In this figure, the data is organized such that a user may select the component of the data they wish to view while hiding the details of the other components. Thus, a user may quickly select the section of data they wish to view and ignore the other sections. For example, as illustrated, a user may select to individually view plan 2002, debrief 2004, or saved configurations 2006. By selecting plan 2002, a user may view, for example, information regarding their answers to the questions regard their plan that they had previously entered. By selecting debrief 2004, a user may view information regarding their answers to questions during a debriefing session and by selecting saved configurations 2006, a user may view the final configurations of the user devices.

Assessment

Using the above described Engine or Virtual Classroom permits assessing a user's proficiency regarding a device by assessing the user/student's control over the device.

Traditionally, students are assessed based on taking a test where they are asked questions regarding the device or control of the device. In an embodiment of the present invention, rather than simply asking a student questions regarding the device, the student is presented with a real world task regarding the control of the device. The student controls the device to perform the task. After completing the task, the student is assessed on his/her performance.

Figure 21 provides an illustration of an assessment engine 2110 that uses a wiretap 2112 to check the configuration of the user devices 2114 once the assignment is completed. As with

24

the wiretap discussed above, various configurations may be used to implement the wiretap 2112. For example, KIBITZ may be used to implement the wiretap 2112. The assessment engine 2110 may be a piece of software running on a processor or computer. Once the assignment is completed, the assessment engine 2110 may examine the configuration of the various user devices 2114 using the wiretap 2112. The assessment engine then checks this configuration against an expected configuration. It then may generate a score. Further, the assessment engine can inform the user of how their configuration differs from the expected configuration, and the potential consequences of these differences.

Figure 22 illustrates an embodiment in which the server and controller 2210 contains an assessment engine 2226. This assessment engine 2226 may be a piece of software running on a processor. At the completion of the assignment, the user's CPE 2212 may be disconnected from the user devices. The assessment engine 2226 then connects to the various user devices 40_1 to 40_3 through the wiretap 2228 of pod controller 24. The assessment engine 2226 then examines the configurations of the user devices 40_1 to 40_3 and compares these configurations with expected configurations. The assessment engine 2226 then may generate a score for the user, and provide him/her with feedback regarding the differences between his/her configurations and the expected configurations. Further, as will be obvious to one of skill in the art, in other embodiments, the assessment engine 2226 may connect to the various devices through a direct connection.

26

Further, as previously discussed in one embodiment a user may be presented with a set of steps that the user can execute to assess his or her final configuration. Referring back, Figure 19 illustrates an example of a page that provides a user with steps for checking their results.

While it has been illustrated and described what is at present considered to be the preferred embodiment and methods of the present invention, it will be understood by those skilled in the art that various changes and modifications may be made, and equivalents may be substituted for elements thereof without departing from the true scope of the invention.

In addition, many modifications may be made to adapt a particular element, technique or, implementation to the teachings of the present invention without departing from the central scope of the invention. Therefore, it is intended that this invention not be limited to the particular embodiment and methods disclosed herein, but that the invention includes all embodiments falling within the scope of the appended claims.